Available online at www.sciencedirect.com
SciVerse ScienceDirect
Procedia CIRP 3 (2012) 525 – 530
www.elsevier.com/locate/procedia

45th CIRP Conference on Manufacturing Systems 2012

Real–Virtual Fusion Production Scheduling Using Social Contract-based Approach – Effectiveness of Adjusting Virtual System Size –

N. Fujii[a]*, Y. Qian[b], T. Kaihara[a]

[a]*Graduate School of System Informatics, Rokkodai 1-1, Nada, Kobe 657-8501, Japan*
[b]*Graduate School of Engineering, Kobe University, Rokkodai 1-1, Nada, Kobe 657-8501, Japan*
* Corresponding author. Tel.: +81-78-803-6133; fax: +81-78-803-6133. *E-mail address*: nfujii@phoenix.kobe-u.ac.jp.

Abstract

This report proposes a new concept of an agent-based Real–Virtual Fusion Manufacturing System (RVF-MS), with the aim of dealing with both external and internal fluctuations adaptively and effectively by realizing a fusion between a real production shop floor and a virtual manufacturing system model. This paper presents a proposal of a social-contract-based production scheduling method in RVF-MS, which generates a virtual system with partial components of the system at the operational phase of production. The influence of the generated range of virtual systems on the rescheduling results is confirmed through computational experiments on flexible flow shop problems.

Keywords: Real–Virtual fusion Manufacturing Systems; Production scheduling; Social contract

1. Introduction

The manufacturing operational phase has many external fluctuations such as order changes and delayed delivery of materials, along with internal uncertain factors such as machine failure and process delay existing on an actual shop floor. Virtual systems designed in advance can not always follow real production situation changes in such complex and dynamic environments exactly. To address both external and internal fluctuations and to obtain proper results to support decision-making, it is important to couple real and virtual systems tightly at the operational phase [1][2]. In the traditional method, virtual systems are usually constructed in advance based on circumstances related to real systems. The parameters of the virtual systems are also modified by collecting actual results that occur on an actual production floor. However, it is difficult to modify only a part of the virtual system such as the cases of machine failure or new machine introduction. A virtual system might be reconstructed again from the beginning in some other cases. A new method that can construct the virtual systems flexibly in the short term to follow changes that occur in real systems is necessary.

This study examines a concept, Real–Virtual Fusion Manufacturing System (RVF-MS), for coupling real and the virtual systems. The proposed RVF-MS are divisible into two parts: the planning phase and operation phase. During the planning phase, a virtual system is constructed by all components in the real system, and decision-making is done through communication of all agents to achieve global production optimization. In turn, at the operational phase, to provide a quick response for dynamic fluctuations, considering that global optimization for the whole manufacturing system is not always necessary, decision-making can be done solely by partial agents that are influenced by the fluctuations. As presented in Figure 1, the RVF-MS brings a new idea of dynamic construction of virtual systems according to the current situation of a real system when new decision-making is necessary. To realize such a fusion between

2212-8271 © 2012 The Authors. Published by Elsevier B.V. Selection and/or peer-review under responsibility of Professor D. Mourtzis and Professor G. Chryssolouris. Open access under CC BY-NC-ND license.
doi:10.1016/j.procir.2012.07.090

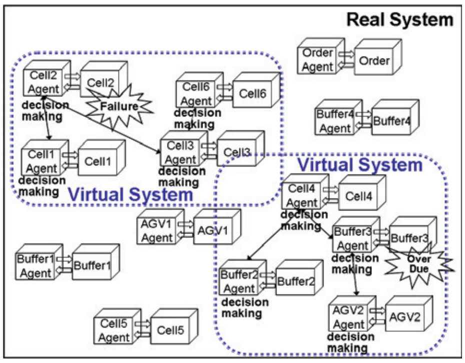

Fig. 1. Concept of RVF-MS

real and the virtual systems, the concept based on multi-agent system [3][4] is applied for constructing a dynamic model in a virtual system. In RVF-MS, agents of two kinds exist: order agents, which generate jobs; and equipment agents such as machine agents and AGV agents. They are modeled as decentralized autonomous agents who can make decisions in the virtual system and who can instruct production activities simultaneously in the real system. These agents in RVF-MS can detect both external fluctuations and internal uncertain factors in the real shop floor and construct virtual models dynamically only when a new decision-making must be done, and instruct production activities according to new results executed in the virtual system. The concept of the RVF-MS presents a new idea that a virtual system is not always necessary, which is used to follow changes of a real system. Furthermore, to provide a quick response to the fluctuations, virtual systems are not always constructed by all components existing in real systems. They can comprise only necessary partial components. This is a different characteristic from those of traditional methods. The virtual system in RVF-MS can be constructed dynamically only when new decision-making is required, and can be destructed when the decision-making is completed. Thereby, the virtual system appears from the real system and disappears into the real system when its mission is completed.

In previous works related to this research, Real–Virtual Fusion Manufacturing Scheduling method is proposed under the RVS-MS concept. To realize the scheduling method, three points described below should be noted:

- Point 1. When to construct a virtual system for decision-making dynamically (rescheduling timing)
- Point 2. How to construct a virtual system for decision-making dynamically (limitation of partial components)
- Point 3. How to solve problems occurring in the real system (Decision-making method in a virtual system)

Information Propagation method [5], based on Recursive Propagation method [6], is proposed for resolving Point 1. The proposed Information Propagation method can forecast the influence of process delays, and can judge whether rescheduling is necessary or not only when tardiness will result from a process delay in the future. For Point 2, to give a quick response to fluctuations without destroying the optimality of an initial schedule, only partial components influenced by Information Propagation are chosen to construct a virtual system by copying their current states from the real system. For Point 3, a social contract-based approach named Combinatorial Auction (CA) [7][8] is applied to determine a new schedule. The balance of execution time and the accuracy of rescheduling can be satisfied by limiting the range of components appropriately and by adjusting the CA parameters [9].

As described in this paper, a method of limiting partial components for generating virtual systems to give a smaller range of decision-making for rapid rescheduling execution (Point 2) is proposed at the operational phase, for the social contract based production scheduling method in RVF-MS. The influence of the generated range of the virtual system on the rescheduling results is elucidated using computational experiments on the flexible flow shop problem.

2. Limitation Method of Components in the Virtual System

Choosing an appropriate range of partial components for generating the virtual systems is important to realize the quick response to the fluctuations in the real system without destroying the optimality of the initial schedule. Based on the result of the Information Propagation method, limitation methods of components can be regarded as the following two steps:

- Step 1. Limitation of equipment agents to participate in the rescheduling
- Step 2. Limitation of target processes for rescheduling, which are collected from participating equipment agents

An Equipment Limitation criterion for limiting the participation of agents (Step 1) is defined: only the equipment agents influenced by the Information Propagation are allowed to participate in the rescheduling. Not only is the participation of equipment agents limited, but the target processes for rescheduling are also limited by two criteria (Step 2) called Time Limitation and Shifted-Process Limitation. Thereby, the range of the generated virtual system can be more restricted. The Time Limitation criterion is defined only as the processes which are finished before the completion period of rescheduling (T_f) are regarded as target processes and only the targeted processes can be reallocated. T_f is a parameter that is definable such as a static value or the finish time of the delayed processes. The Shifted-Process Limitation criterion is defined only as the processes which are influenced by Information Propagation. The planned processing start time and finish time, which are shifted to the right (or left), are regarded as target processes. Furthermore, the combination of Time Limitation and Shifted-Process Limitation can be regarded as a limit into the smaller range of the target processes for rescheduling.

Virtual systems are generated by the combination of limitation criteria. Therefore, the range of problems can be adjusted. *VS* is definable as a virtual system presented in Equation (1). Virtual systems can be generated as eight models, shown as Equations (2)–(9) by the combination of participating equipment and target processes for rescheduling. Nomenclature used in the paper is presented in Table 1.

Table 1. Nomenclature

j	job number
k	process number
$J_{j,k}$	process of job j in process k
m	equipment
$st_{j,k}$	start time of job j in process k after Information Propagation
$ft_{j,k}$	finish time of job j in process k after Information Propagation
$st^*_{j,k}$	start time of job j in process k before Information Propagation
$ft^*_{j,k}$	finish time of job j in process k before Information Propagation
T_{now}	present time of real system (finish time of Information Propagation)
T_r	expected execution time of rescheduling
T_s	start period of rescheduling
T_f	finish period of rescheduling
F_{ID}	set of all participating equipment influenced by Information Propagation
SM_r	set of all participating equipment for rescheduling
SJ_r	set of all target processes for rescheduling collected from all participating equipment in SM_r

$$VS = < SM_r, SJ_r > \quad (1)$$

SM_r: set of all participating equipment for rescheduling
SJ_r: set of all target processes for rescheduling collected from all the participating equipment in SM_r

<u>*VS1*: No Limitation</u>
No limitation either in participating equipment or the target processes for rescheduling, as

$$VS1 = < SM_r, SJ_r >, \quad (2)$$
$$\text{where } SM_r = \{ m \mid {}^{\forall}m \}$$
$$SJ_r = \{ J_{j,k} \mid st_{j,k} \geq T_s \}.$$

<u>*VS2*: Time Limitation</u>
No limitation in the participating equipment. The target processes for rescheduling are limited by the Time Limitation criterion as

$$VS2 = < SM_r, SJ_r > , \quad (3)$$
$$\text{where } SM_r = \{ m \mid {}^{\forall}m \}$$
$$SJ_r = \{ J_{j,k} \mid st_{j,k} \geq T_s \text{ and } ft_{j,k} \leq T_f \}.$$

<u>*VS3*: Shift-Process Limitation</u>
No limitation in the participating equipment. The target processes for rescheduling are limited by Shift-Process Limitation criterion, as

$$VS3 = < SM_r, SJ_r >, \quad (4)$$
$$\text{where } SM_r = \{ m \mid {}^{\forall}m \}$$
$$SJ_r = \{ J_{j,k} \mid st_{j,k} \geq T_s \text{ and } st_{j,k} \neq st^*_{j,k} \}.$$

<u>*VS4*: Time and Shift-Process Limitations</u>
No limitation in the participating equipment. Target processes for rescheduling are limited by the combination of Time Limitation and Shift-Process Limitation criteria, as

$$VS4 = < SM_r, SJ_r > , \quad (5)$$
$$\text{where } SM_r = \{ m \mid {}^{\forall}m \}$$
$$SJ_r = \{ J_{j,k} \mid st_{j,k} \geq T_s \text{ and } ft_{j,k} \leq T_f \text{ and } st_{j,k} \neq st^*_{j,k} \}.$$

<u>*VS5*: Equipment Limitation</u>
Participating equipment is limited by the Equipment Limitation criterion. No limitation in the target processes for rescheduling applies, as

$$VS5 = < SM_r, SJ_r >, \quad (6)$$
$$\text{where } SM_r = \{ m \mid m \in F_{ID} \}$$
$$SJ_r = \{ J_{j,k} \mid st_{j,k} \geq T_s \}.$$

<u>*VS6*: Equipment and Time Limitations</u>
Participating equipment is limited by Equipment Limitation criterion. Target processes for rescheduling are limited by Time Limitation criterion.

$$VS6 = < SM_r, SJ_r > \quad (7)$$
$$\text{Therein, } SM_r = \{ m \mid m \in F_{ID} \}$$
$$SJ_r = \{ J_{j,k} \mid st_{j,k} \geq T_s \text{ and } ft_{j,k} \leq T_f \}.$$

<u>*VS7*: Equipment and Shift-Process Limitations</u>
Participating equipment is limited by the Equipment Limitation criterion. Target processes for rescheduling are limited by the Shift-Process Limitation criterion.

$$VS7 = < SM_r, SJ_r > \quad (8)$$
$$\text{In that equation, } SM_r = \{ m \mid m \in F_{ID} \}$$

$SJ_r = \{ J_{j,k} \mid st_{j,k} \geq T_s \text{ and } st_{j,k} \neq st^*_{j,k} \}.$

<u>*VS8*: Equipment, Time and Shift-Process Limitations</u>

Participating equipment is limited by Equipment Limitation standard. Target processes for rescheduling are limited by the combination of Time Limitation and Shift-Process Limitation criteria as

$$VS8 = < SM_r, SJ_r >, \quad (9)$$

where $SM_r = \{ m \mid m \in F_{ID} \}$

$SJ_r = \{ J_{j,k} \mid st_{j,k} \geq T_s \text{ and } ft_{j,k} \leq T_f \text{ and } st_{j,k} \neq st^*_{j,k} \}.$

VS1–VS4 are generated only by considering the target processes for rescheduling without considering the limitation of participating equipment agents. *VS5–VS8* are generated with consideration of the limitation of participating equipment agents. *VS8* has the smallest range of the virtual system. It can be regarded as the shortest execution time of rescheduling, but the accuracy of rescheduling results might drop because it has the smallest range for modification of the initial schedule. However, *VS1* is generated with the largest range of the virtual system and it can attain the highest accuracy of rescheduling results. Nevertheless, it might entail the longest execution time for rescheduling.

Figure 2 presents an example of *VS8 to* explain the proposed limitation method in a flexible flow shop problem. M1 and M2 are set to the equipment of Process 1. In turn, M3 and M4 are used as the equipment of Process 2. In this situation, Information Propagation is promoted by M1 at time T_{now} because of a delay that occurred in its J1 process. The delay information is propagated to the downstream processes, as shown by dotted arrows, and tardiness in M3 is forecasted. M1 decides that rescheduling is necessary to modify the initial schedule after the beginning period of rescheduling $T_s = T_{now} + T_r$. A virtual system will be generated by the *VS8* model, which is considered all limitation standards. Participating equipment agents are limited only to M1, M3, and M4, which are influenced by Information Propagation (marked with filled circles). The target processes of rescheduling are limited only to the right-shifted processes before time T_f (marked with filled triangles).

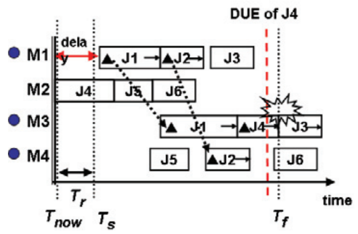

Fig. 2. Example of *VS8*

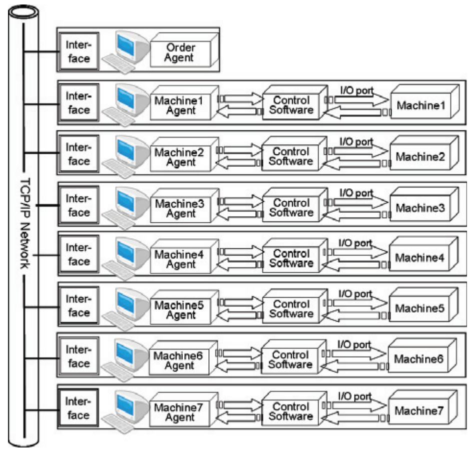

Fig. 3. RVF-MS configuration on a Three-process Flexible Flow Shop Problem

3. Computational Experiments

3.1. Experimental settings

Computational experiments on a Three-process Flexible Flow Shop Problem were executed to assess the proposed method. The system configuration of the RVF-MS is depicted as Figure 3. Machines 1 and 2 are used in Process 1, Machines 3, 4, and 5 are used in Process 2, and Machines 6 and 7 are used in Process 3. All machine agents and order agents can mutually communicate using a TCP/IP network. Decision-making is executable through their mutual communication.

In the experiment, a planning period is set to one day, in which jobs of four kinds (Types A, B, C, and D) are scheduled at the planning phase without tardiness using CA. Processing times of each job are set as shown in Table 2. Two order patterns are shown below. Pattern 2 has more jobs to be processed than Pattern 1 has.

- Pattern 1: A, B, C, and D are respectively ordered 3, and the due times of all jobs are set to 2.5 times of each total processing time.
- Pattern 2: A, B, C, and D are respectively ordered 4, and the due times of all jobs are set to 2.8 times of each total processing time.

Table 2. Processing time (min)

Type	Process 1	Process 2	Process 3	Total time
A	100	50	100	250
B	50	100	50	200
C	50	50	50	150
D	100	100	100	300

Table 3. Comparison of different limitation methods on Pattern 1 (Avg.)

VS model	$\sum T\_Tar$ (min)	$\sum T\_Res$ (s)	$\sum T\_N$	T_Res (s)
VS 1	296.0	53.5	10.0	5.4
VS 2	321.1	43.1	10.0	4.3
VS 3	345.2	43.9	10.0	4.4
VS 4	356.9	45.7	10.7	4.2
VS 5	325.6	41.3	10.4	4.0
VS 6	350.8	34.1	10.6	3.2
VS 7	354.7	39.2	11.0	3.6
VS 8	367.1	34.7	11.2	3.1
No_Res	624.1	0	0	0

Table 4. Comparison of different limitation methods on Pattern 2 (Avg.)

VS model	$\sum T\_Tar$ (min)	$\sum T\_Res$ (s)	$\sum T\_N$	T_Res (s)
VS 1	781.7	691.5	19.7	35.1
VS 2	838.3	580.5	20.8	27.9
VS 3	828.0	599.4	20.2	29.7
VS 4	848.2	494.1	21.4	23.1
VS 5	814.4	559.0	20.4	27.4
VS 6	849.8	519.2	21.1	24.6
VS 7	846.1	477.1	20.5	23.3
VS 8	865.0	452.6	21.8	20.8
No_Res	1281.0	0	0	0

At the planning phase, an initial schedule is generated to achieve global production optimization by all components existing in the real system (all processes and from Machine 1 to Machine 7) using CA method [8]. At the operational phase of the production activities, process delays occur to each process by random values U (20–30) (min), and Information Propagation method [5] is used to forecast whether tardiness will occur in the future. Rescheduling is executed only when the tardiness is forecasted, and the virtual system for decision-making will be generated only by partial components using the limited methods (*VS1–VS8*) proposed in section 2 to give a partial modification to the initial schedule in the short term. Combinatorial auctions with an all bid method [8] will be used as the decision-making method in the virtual system, and the objective function of the system is minimizing the total tardiness of processing jobs. Moreover, for simplifying the problem, execution time T_r of rescheduling is not considered in these experiments, i.e., the beginning period of rescheduling T_s is set to T_{now}.

3.2. Results and discussion

Results of the experiments (averages of 10) are presented in Table 3 and Table 4. The meanings of respective performance indicators in the line direction show results of *VS1–VS8* and the situation with no rescheduling executed (*No_Res*). The meanings of respective performance indicators in the column-direction show the total tardiness when all processing jobs are finished ($\sum T\_Tar$ (min)), the total rescheduling execution time ($\sum T\_Res$ (s)), the total execution times of rescheduling ($\sum T\_N$), and the average execution time of rescheduling for once ($T\_Res = \sum T\_Res \,/\, \sum T\_N$ (s)), respectively. Moreover, from the perspectives of the results of total tardiness $\sum T\_Tar$ and average execution time for rescheduling T_Res, the percent of change (PCT_i) compared to *VS1* of each limitation method is calculated using Equation (10), and the comparison results are portrayed in Figure 4 and Figure 5. VAL_i in Equation (10) presents resultant values of the respective performance indicator methods described above in limitation method VS_i (i=1,…,8).

$$PCT_i = \frac{VAL_i - VAL_1}{VAL_1} \times 100\% \quad (i = 1, \cdots, 8) \tag{10}$$

Regarding results for Pattern 1 shown in Table 3, all limitation methods from *VS1–VS8* yield smaller total tardiness ($\sum T\_Tar$) than the situation with no rescheduling executed (*No_Res*). Compared with the results of *VS1–VS8*, it is readily apparent that the virtual system with a smaller range obtains greater total tardiness ($\sum T\_Tar$) and also obtains higher frequency of rescheduling ($\sum T\_N$). The reason is that the modification range of the initial schedule becomes smaller along with the limitation of a virtual system. The smaller virtual system produces worse accuracy of rescheduling. Results show that the value of total tardiness ($\sum T\_Tar$) and total times of rescheduling ($\sum T\_N$) become larger along with reduced size of the virtual system in the experiments. However, the average execution time of rescheduling (T_Res) becomes shorter along with reduction in the virtual system, which shows the effectiveness of the proposed method by limiting the range of the virtual system in the point of calculation time. Results of *VS1* show that it has the smallest value of total tardiness $\sum T\_Tar$, but that it costs the longest time for rescheduling T_Res. With the smallest range of *VS8*, the shortest rescheduling times *T_Res are obtainable* by saving the time for decision-making, although it decreases the performance of total tardiness. Compared to results presented in Figure 4, although *VS8* becomes about 24% worse on the value of total tardiness

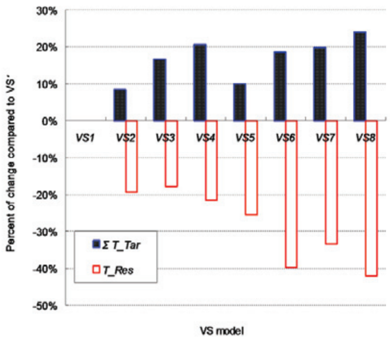

Fig. 4. Comparison of different limitation methods with change of *VS1* on Pattern 1

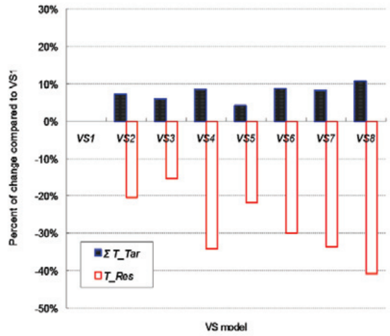

Fig. 5. Comparison of different limitation methods with change of *VS1* on Pattern 2

$\sum T\_Tar$ than *VS1*, it succeeded in saving rescheduling time *T_Res* at about 42%.

Regarding results for Pattern 2 shown in Table 4 and Figure 5, a similar tendency is apparent for total tardiness $\sum T\_Tar$ and the rescheduling time *T_Res* as Pattern 1, for which the balance of $\sum T\_Tar$ and *T_Res* can be realized by limiting the range of virtual systems. Compared with Pattern 1, Pattern 2 acquires about twice the total rescheduling times $\sum T\_N$, 7 times the rescheduling time *T_Res*, and 10 times the total rescheduling execution time $\sum T\_Res$ of Pattern 1. Pattern 2 is a more complicated production situation than Patten 1. Using the proposed method, *VS8* saved rescheduling time *T_Res* of about 41%, although *VS8* yields an approximately 11% worse value of total tardiness $\sum T\_Tar$ than *VS1*. Pattern 2 shows great improvement over Pattern 1 by comparison of the results of the increasing proportion on total tardiness $\sum T\_Tar$ and decreasing proportion on rescheduling time *T_Res*.

Based on the results described above, the effectiveness of the proposed method is verified. The balance between the accuracy and the execution time of rescheduling can be achieved by adjusting the range of generating virtual systems. Improvement of saving the rescheduling time can be expected in a more complicated production situation such as Pattern 2. Moreover, *VS5* probably has the best results for both Pattern 1 and 2 in the experiments by comparing the results of the increasing proportion on total tardiness $\sum T\_Tar$ and decreasing proportion on once rescheduling time *T_Res*. However, the Time limitation criterion does not work effectively in experiments because the modification period of the initial schedule is only set to one day. The Time limitation criterion effectiveness can be expected to be greater at the situation of a long period on the modification of the initial schedule.

4. Conclusion

As described in this paper, a method to limit participating components for generation of the virtual system to give the smaller range of decision-making for agile rescheduling execution is proposed. It conducts the social contract-based production scheduling method in a Real–Virtual Fusion Manufacturing System. Computational experiments related to the flexible flow shop problem verified that the balance of accuracy and execution time of rescheduling can be achieved by adjusting the generated range of the virtual system. It can be expected that the decision-making can be executed in a shorter time without greatly decreased accuracy if the range of virtual systems could be adjusted appropriately.

References

[1] Hibino, H., Inukai, T. Fukuda, Y., 2006. Efficient Manufacturing System Implementation Based on Combination between Real and Virtual Factory, International Journal of Production Research 44, 18-19, p.3897.

[2] Kadar, B., Lengyel, A., Monostori, L. Suginishi, Y., Pfeiffer, A., Nonaka, Y., 2010. Enhanced Control of Complex Production Structures by Tight Coupling of the Digital and the Physical Worlds, CIRP Annals – Manufacturing Technology 59, 1, p.437.

[3] Papakostas, N., Mourtzis, D., Michalos, G., Makris, S., Chryssolouris, G., 2011. An agent-based methodology for manufacturing decision making: a textile case study, International Journal of Computer Integrated Manufacturing, DOI: 10.1080/0951192X.2011.637963.

[4] Monostori, L., Vancza, J., Kumara, S.R.T., 2006. Agent-based systems for manufacturing, CIRP Annals – Manufacturing Technology 55, 2, p. 697.

[5] Qian, Y., Fujii, N., Kaihara, T., Fujii, S., Umeda, T., 2011, Dynamic Construction of Virtual System in Real–Virtual Fusion Manufacturing System, Systems Control and Information 24, 2, p.31.

[6] Hino, R., Moriwaki, T., 2004, Decentralized Job Shop Scheduling by Recursive Propagation Method, JSME International Journal Series C 45, 2, p. 551.

[7] Kaihara, T., Fujii, N., Toide, S., Ishibashi, H., 2009, A study on Optimization Method for Production Scheduling with Batch Process by Combinatorial Auction, International Symposium on Scheduling 2009, p. 266.

[8] Qian, Y., Fujii, N., Kaihara, T., Fujii, S., Umeda, T., 2011, Production Scheduling with Social Contract Based Approach for Real–Virtual Fusion Manufacturing System, 44th CIRP Conference on Manufacturing Systems, CD-ROM.

[9] Qian, Y., Fujii, N., Kaihara, T., Fujii, S., Umeda, T., 2011, Scheduling Method in Real–Virtual Fusion Manufacturing System with Social Contract Based Approach -Application of Combinatorial Auction at Operational Phase, International Symposium on Scheduling 2011, p.57-62.